REFORMULATION OF TWO STEP IMPLICIT LINEAR MULTI-STEP BLOCK HYBRID METHOD INTO RUNGE –KUTTA TYPE METHOD FOR THE SOLUTION OF SECOND ORDER INITIAL VALUE PROBLEM (IVP)

[1]ALIYU Abubakar

[2]MUHAMMAD Raihanatu

[3]ABDULHAKEEM Yusuf

[1,2&3]Department of Mathematics, School of Physical Sciences, Federal University of Technology Minna, Niger State, Nigeria

Abstract

Second-order ordinary differential equations (ODEs) is unavoidable in scientific and engineering fields. This research focuses on the reformulation of two-step implicit linear multistep block hybrid method into a seven-stage Runge-Kutta type method for the solution of second-order initial value problems (IVPs). A two-step, four-off-grid-point implicit block hybrid collocation method for first-order initial value problems was derived. Its order and error constants were determined, which shows that the schemes were of order 8, 8, 8, 8, 8 and 9 with respective error constants of $\frac{275}{158723712}, \frac{8}{6200145}, \frac{1}{653184}, \frac{8}{6200145}, \frac{275}{158723712}$, and $-\frac{1}{3061800}$. The derived block method was reformulated into a seven-stage Runge-Kutta type method (RKTM) for the solution of first-order ordinary differential equations; this reformulation was extended to handle the required second-order ordinary differential equations. The second-order Runge-Kutta-type method derived was implemented on numerical experiments. The method was found to be better than existing methods in the literature.

Keywords: Implicit, Hybrid block method, collocation, Runge-Kutta type method, linear multistep method.

1.0 Introduction

The accurate numerical solution of second-order initial value problems (IVPs) in ordinary differential equations (ODEs) remains a significant challenge in scientific and engineering computations (Datsko and Kutniv, 2024). These problems are mostly expressed in the general form:

$$y''(x) = f(x, y, y'),\ y(x_0) = y_0, y'(x_0) = y'_0 \tag{1}$$

that arise in several fields such as applied mathematics, physics and engineering (Denis, 2020). Conventional methods always reduce second-order ODEs into systems of first-order equations, hence it can cause additional computational burden and inefficiencies (Li *et al.*, 2024). To tackle these challenges, linear multi-step methods (LMMs) together with the hybrid and block scheme have been developed for direct solution of second-order ODEs (Olaleye and Fiyinfoluwa, 2023; Taıwo *et al.*, 2023).

Block hybrid linear multi-step methods are known for their ability to compute solutions at multiple points within single step, improving it multi solution processing ability and reducing approximating errors (Li *et al.*, 2024; Omole *et al.*, 2024). However, these methods mostly depend on already-computed starting values and cannot withstand variable step sizes (Sharma, 2021). Reformulating these methods into the Runge–Kutta Type method give an avenue to by-pass these challenges by bringing in a self-starting property and better adaptability to changing step sizes.

This work focuses on reformulating a two-step block hybrid linear multi-step method into a Runge–Kutta-Type scheme for directly solving second-order IVPs. By utilizing the adaptability of Runge–Kutta methods and the efficiency of hybrid block methods, the new proposed techniques seek to improve the numerical performance and flexibility of finding the solution of second-order ODEs. The theoretical analysis of the reformulated method, including its stability and convergence properties, is complemented by numerical experiments to demonstrate its effectiveness and accuracy. This study complement the advancement in the numerical methods for solving IVPs, offering a robust and versatile tool for current computational challenges.

2.0 Materials and Methods

The derivation of Seven – Stage Runge – Kutta Type method reformulated from two step block hybrid linear multistep method with four intra-step points is discussed in this section. First order two step block hybrid linear multistep method with four off-grids point was reformulated to first order Runge - Kutta Type method.

In general, initial value problem (IVP) in first Order Differential Equations (ODEs) is of the form of form:

$$y'(x) = f(x, y),\ y(x_0) = y_0 \tag{2}$$

We assume power series of the form:

$$y(x) = \sum_{i=0}^{l-1} a_i x^i, l = p + c$$

(3) Differentiating (3) gives:

$$y^1(x) = \sum_{i=0}^{l-1} i a_i x^{i-1} \tag{4}$$

We find $a_i's$ using matrix inversion techniques to obtain a continuous implicit scheme of the form:

$$y(x) = \alpha_0(x)y_n + h\left(\beta_{n+\frac{1}{3}}f_{n+\frac{1}{3}} + \beta_{n+\frac{2}{3}}f_{n+\frac{2}{3}} + \beta_{n+\frac{4}{3}}f_{n+\frac{4}{3}} + \beta_{n+\frac{5}{3}}f_{n+\frac{5}{3}} + \left(\sum_{i=0}^{2}\beta_i(x)f_{n+i}\right)\right)$$

(5)

Interpolating (3) at $x=\{x_n\}$ and collocating (4) at $x=\left\{x_n, x_{n+\frac{1}{3}}, x_{n+\frac{2}{3}}, x_{n+1}, x_{n+\frac{4}{3}}, x_{n+\frac{5}{3}}, x_{n+2}\right\}$.

It can be expressed in matrix form as:

$$\begin{bmatrix} 1 & x_n & x_n^2 & x_n^3 & x_n^4 & x_n^5 & x_n^6 & x_n^7 \\ 0 & 1 & 2x_n & 3x_n^2 & 4x_n^3 & 5x_n^4 & 6x_n^5 & 7x_n^6 \\ 0 & 1 & 2x_{n+\frac{1}{3}} & 3(x_{n+\frac{1}{3}})^2 & 4(x_{n+\frac{1}{3}})^3 & 5(x_{n+\frac{1}{3}})^4 & 6x_{n+\frac{1}{3}})^5 & 7x_{n+\frac{1}{3}})^6 \\ 0 & 1 & 2x_{n+\frac{2}{3}} & 3(x_{n+\frac{2}{3}})^2 & 4(x_{n+\frac{2}{3}})^3 & 5(x_{n+\frac{2}{3}})^4 & 6(x_{n+\frac{2}{3}})^5 & 7(x_{n+\frac{2}{3}})^6 \\ 0 & 1 & 2x_{n+1} & 3(x_{n+1})^2 & 4(x_{n+1})^3 & 5(x_{n+1})^4 & 6(x_{n+1})^5 & 7(x_{n+1})^6 \\ 0 & 1 & 2x_{n+\frac{4}{3}} & 3(x_{n+\frac{4}{3}})^2 & 4(x_{n+\frac{4}{3}})^3 & 5(x_{n+\frac{4}{3}})^4 & 6(x_{n+\frac{4}{3}})^5 & 7(x_{n+\frac{4}{3}})^6 \\ 0 & 1 & 2x_{n+\frac{5}{3}} & 3(x_{n+\frac{5}{3}})^2 & 4(x_{n+\frac{5}{3}})^3 & 5(x_{n+\frac{5}{3}})^4 & 6(x_{n+\frac{5}{3}})^5 & 7(x_{n+\frac{5}{3}})^6 \\ 0 & 1 & 2x_{n+2} & 3(x_{n+2})^2 & 4(x_{n+2})^3 & 5(x_{n+2})^4 & 6(x_{n+2})^5 & 7(x_{n+2})^6 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix} = \begin{bmatrix} y_n \\ hf_n \\ hf_{n+\frac{1}{3}} \\ hf_{n+\frac{2}{3}} \\ hf_{n+1} \\ hf_{n+\frac{4}{3}} \\ hf_{n+\frac{5}{3}} \\ hf_{n+2} \end{bmatrix} \quad (6)$$

Using Maple Mathematical software we find the values of $a's$ in (5) which gave the continuous formula

$$y(x) = \alpha_0(x)y_n + h\left[\beta_n f_n + \beta_{n+\frac{1}{3}}f_{n+\frac{1}{3}} + \beta_{n+\frac{2}{3}}f_{n+\frac{2}{3}} + \beta_{n+1}f_{n+1} + \beta_{n+\frac{4}{3}}f_{n+\frac{4}{3}} + \beta_{n+\frac{5}{3}}f_{n+\frac{5}{3}} + \beta_{n+2}f_{n+2}\right]$$

(7)

Taking $t = \frac{x-x_{n+1}}{h} \Rightarrow \frac{dt}{dx} = \frac{1}{h}$ (8)

Evaluating (6) at $x = x_{n+1}, i = \frac{1}{3}, \frac{2}{3}, 1, \frac{4}{3}, \frac{5}{3}, and$ 2 yields discrete implicit block method

$$\left.\begin{aligned}
y_{n+2} &= y_n + h\left(\frac{41}{420}f_n + \frac{18}{35}f_{n+\frac{1}{3}} + \frac{9}{140}f_{n+\frac{2}{3}} + \frac{68}{105}f_{n+1} + \frac{9}{140}f_{n+\frac{4}{3}} + \frac{18}{35}f_{n+\frac{5}{3}} + \frac{41}{420}f_{n+2}\right) \\
y_{n+\frac{5}{3}} &= y_n + h\left(\frac{3715}{36288}f_n + \frac{725}{1512}f_{n+\frac{1}{3}} + \frac{2125}{12096}f_{n+\frac{2}{3}} + \frac{250}{567}f_{n+1} + \frac{3875}{12096}f_{n+\frac{4}{3}} + \frac{235}{1512}f_{n+\frac{5}{3}} - \frac{275}{36288}f_{n+2}\right) \\
y_{n+\frac{4}{3}} &= y_n + h\left(\frac{286}{2835}f_n + \frac{464}{945}f_{n+\frac{1}{3}} + \frac{128}{945}f_{n+\frac{2}{3}} + \frac{1504}{2835}f_{n+1} + \frac{58}{945}f_{n+\frac{4}{3}} + \frac{16}{945}f_{n+\frac{5}{3}} - \frac{8}{2835}f_{n+2}\right) \\
y_{n+1} &= y_n + h\left(\frac{137}{1344}f_n + \frac{27}{56}f_{n+\frac{1}{3}} + \frac{387}{2240}f_{n+\frac{2}{3}} + \frac{34}{105}f_{n+1} - \frac{243}{2240}f_{n+\frac{4}{3}} + \frac{9}{280}f_{n+\frac{5}{3}} - \frac{29}{6720}f_{n+2}\right) \\
y_{n+\frac{2}{3}} &= y_n + h\left(\frac{1139}{11340}f_n + \frac{94}{189}f_{n+\frac{1}{3}} + \frac{11}{3780}f_{n+\frac{2}{3}} + \frac{332}{2835}f_{n+1} - \frac{269}{3780}f_{n+\frac{4}{3}} + \frac{22}{945}f_{n+\frac{5}{3}} - \frac{37}{11340}f_{n+2}\right) \\
y_{n+\frac{1}{3}} &= y_n + h\left(\frac{19087}{181440}f_n + \frac{2713}{7560}f_{n+\frac{1}{3}} - \frac{15487}{60480}f_{n+\frac{2}{3}} + \frac{586}{2835}f_{n+1} - \frac{6737}{60480}f_{n+\frac{4}{3}} + \frac{263}{7560}f_{n+\frac{5}{3}} - \frac{863}{181440}f_{n+2}\right)
\end{aligned}\right\}$$

(9)

Arranging the discrete scheme (8) in matrix equation form:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_{n+\frac{1}{3}} \\ y_{n+\frac{2}{3}} \\ y_{n+1} \\ y_{n+\frac{4}{3}} \\ y_{n+\frac{5}{3}} \\ y_{n+2} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_{n-\frac{5}{3}} \\ y_{n-\frac{4}{3}} \\ y_{n-1} \\ y_{n-\frac{2}{3}} \\ y_{n-\frac{1}{3}} \\ y_n \end{bmatrix}$$

$$+\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \frac{19087}{181440} \\ 0 & 0 & 0 & 0 & 0 & \frac{1139}{11340} \\ 0 & 0 & 0 & 0 & 0 & \frac{137}{1344} \\ 0 & 0 & 0 & 0 & 0 & \frac{286}{2835} \\ 0 & 0 & 0 & 0 & 0 & \frac{3715}{36288} \\ 0 & 0 & 0 & 0 & 0 & \frac{41}{4200} \end{bmatrix} \begin{bmatrix} f_{n-\frac{5}{3}} \\ f_{n-\frac{4}{3}} \\ f_{n-1} \\ f_{n-\frac{2}{3}} \\ f_{n-\frac{1}{3}} \\ f_n \end{bmatrix} + \begin{bmatrix} \frac{2713}{7560} & \frac{-15487}{60480} & \frac{586}{2835} & \frac{-6737}{60480} & \frac{263}{7560} & \frac{-863}{181440} \\ \frac{94}{189} & \frac{11}{3780} & \frac{332}{2835} & \frac{-269}{3780} & \frac{22}{945} & \frac{-37}{11340} \\ \frac{27}{56} & \frac{387}{2240} & \frac{34}{105} & \frac{-243}{2240} & \frac{9}{280} & \frac{-29}{6720} \\ \frac{464}{945} & \frac{128}{945} & \frac{1504}{2835} & \frac{58}{945} & \frac{16}{945} & \frac{-8}{2835} \\ \frac{725}{1512} & \frac{2125}{12096} & \frac{250}{567} & \frac{3875}{12096} & \frac{235}{1512} & \frac{-275}{36288} \\ \frac{18}{35} & \frac{9}{140} & \frac{68}{105} & \frac{9}{140} & \frac{18}{35} & \frac{41}{420} \end{bmatrix} \begin{bmatrix} f_{n+\frac{1}{3}} \\ f_{n+\frac{2}{3}} \\ f_{n+1} \\ f_{n+\frac{4}{3}} \\ f_{n+\frac{5}{3}} \\ f_{n+2} \end{bmatrix}$$

(10)

2.1 Analysis of the Block Method

2.1.1 Order and error constant of the block method

We discuss the approximation of the order and error constant of the block using the difference equation:

$$L[y(x), h] = \sum_{i=0}^{k}[\alpha_i y(x + ih) - h\beta y'(x + ih)]$$

(11)

If we assume that $y\ (x)$ has as many higher derivatives as we require, we can expand the terms in (11) as a Taylor series about the point x to obtain the expansion

$$L[y(x), h] = C_O y(x) + C_1 h y^{(1)}(x) + \cdots + C_q h^q y^q(x) + \cdots \quad (12)$$

Where

$$\left.\begin{aligned} C_0 &= \alpha_0 + \alpha_1 + \cdots + \alpha_k \\ C_1 &= (\alpha_1 + \alpha_2 + \cdots + k\alpha_k) - (\beta_0 + \cdots + k^{(q-1)}\beta_k \\ &. \\ &. \\ &. \\ C_q &= \frac{1}{q!}(\alpha_1 + \alpha_2 + \cdots + k^q\alpha_k) - \frac{1}{(q-1)!}(\beta_1 + \cdots + k^{(q-1)}\beta_k \end{aligned}\right\} \quad (13)$$

Where q = 2, 3, …, k

For $y_{n+2} = y_n + h\left(\frac{41}{420}f_n + \frac{18}{35}f_{n+\frac{1}{3}} + \frac{9}{140}f_{n+\frac{2}{3}} + \frac{68}{105}f_{n+1} + \frac{9}{140}f_{n+\frac{4}{3}} + \frac{18}{35}f_{n+\frac{5}{3}} + \frac{41}{420}f_{n+2}\right)$

The order and error constant is obtained using these coefficient:

$\alpha_0 = -1, \alpha_2 = 1, \beta_0 = \frac{41}{420}, \beta_{\frac{1}{3}} = \frac{18}{35}, \beta_{\frac{2}{3}} = \frac{9}{140}, \beta_1 = \frac{68}{105}, \beta_{\frac{4}{3}} = \frac{9}{140}, \ \beta_{\frac{5}{3}} = \frac{18}{35}, \beta_2 = \frac{41}{420}$

The computation is given in the form:

$$\begin{bmatrix} C_0 \\ C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \\ C_7 \\ C_8 \\ C_9 \end{bmatrix} = \begin{bmatrix} \alpha_2 + \alpha_0 \\ 2\alpha_2 - (\beta_0 + \beta_{\frac{1}{3}} + \beta_{\frac{2}{3}} + \beta_1 + \beta_{\frac{4}{3}} + \beta_{\frac{5}{3}} + \beta_2) \\ \frac{1}{2!}2^2\alpha_2 - \frac{1}{1!}\left(\frac{1}{3}\beta_{\frac{1}{3}} + \frac{2}{3}\beta_{\frac{2}{3}} + \beta_1 + \frac{4}{3}\beta_{\frac{4}{3}} + \frac{5}{3}\beta_{\frac{5}{3}} + 2\beta_2\right) \\ \frac{1}{3!}2^3\alpha_2 - \frac{1}{2!}\left(\left(\frac{1}{3}\right)^2\beta_{\frac{1}{3}} + \left(\frac{2}{3}\right)^2\beta_{\frac{2}{3}} + \beta_1 + \left(\frac{4}{3}\right)^2\beta_{\frac{4}{3}} + \left(\frac{5}{3}\right)^2\beta_{\frac{5}{3}} + 2^2\beta_2\right) \\ \frac{1}{4!}2^4\alpha_2 - \frac{1}{3!}\left(\left(\frac{1}{3}\right)^3\beta_{\frac{1}{3}} + \left(\frac{2}{3}\right)^3\beta_{\frac{2}{3}} + \beta_1 + \left(\frac{4}{3}\right)^3\beta_{\frac{4}{3}} + \left(\frac{5}{3}\right)^3\beta_{\frac{5}{3}} + 2^3\beta_2\right) \\ \frac{1}{5!}2^5\alpha_2 - \frac{1}{4!}\left(\left(\frac{1}{3}\right)^4\beta_{\frac{1}{3}} + \left(\frac{2}{3}\right)^4\beta_{\frac{2}{3}} + \beta_1 + \left(\frac{4}{3}\right)^4\beta_{\frac{4}{3}} + \left(\frac{5}{3}\right)^4\beta_{\frac{5}{3}} + 2^4\beta_2\right) \\ \frac{1}{6!}2^6\alpha_2 - \frac{1}{5!}\left(\left(\frac{1}{3}\right)^5\beta_{\frac{1}{3}} + \left(\frac{2}{3}\right)^5\beta_{\frac{2}{3}} + \beta_1 + \left(\frac{4}{3}\right)^5\beta_{\frac{4}{3}} + \left(\frac{5}{3}\right)^5\beta_{\frac{5}{3}} + 2^5\beta_2\right) \\ \frac{1}{7!}2^7\alpha_2 - \frac{1}{6!}\left(\left(\frac{1}{3}\right)^6\beta_{\frac{1}{3}} + \left(\frac{2}{3}\right)^6\beta_{\frac{2}{3}} + \beta_1 + \left(\frac{4}{3}\right)^6\beta_{\frac{4}{3}} + \left(\frac{5}{3}\right)^6\beta_{\frac{5}{3}} + 2^6\beta_2\right) \\ \frac{1}{8!}2^8\alpha_2 - \frac{1}{7!}\left(\left(\frac{1}{3}\right)^7\beta_{\frac{1}{3}} + \left(\frac{2}{3}\right)^7\beta_{\frac{2}{3}} + \beta_1 + \left(\frac{4}{3}\right)^7\beta_{\frac{4}{3}} + \left(\frac{5}{3}\right)^7\beta_{\frac{5}{3}} + 2^7\beta_2\right) \\ \frac{1}{9!}2^9\alpha_2 - \frac{1}{8!}\left(\left(\frac{1}{3}\right)^8\beta_{\frac{1}{3}} + \left(\frac{2}{3}\right)^8\beta_{\frac{2}{3}} + \beta_1 + \left(\frac{4}{3}\right)^8\beta_{\frac{4}{3}} + \left(\frac{5}{3}\right)^8\beta_{\frac{5}{3}} + 2^8\beta_2\right) \end{bmatrix} =$$

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\frac{1}{3061800} \end{bmatrix} \qquad (14)$$

Using similar procedures to obtain the order and error constants of the schemes (8).

Table 2.1: Order and error constants of the scheme (8)

Scheme	Order	Error Constant
$y_{n+\frac{1}{3}} = y_n + h\left(\frac{19087}{181440}f_n + \frac{2713}{7560}f_{n+\frac{1}{3}} - \frac{15487}{60480}f_{n+\frac{2}{3}} + \frac{586}{2835}f_{n+1} - \frac{6737}{60480}f_{n+\frac{4}{3}} + \frac{263}{7560}f_{n+\frac{5}{3}} - \frac{863}{181440}f_{n+2}\right)$	8	$\frac{275}{158723712}$
$y_{n+\frac{2}{3}} = y_n + h\left(\frac{1139}{11340}f_n + \frac{94}{189}f_{n+\frac{1}{3}} + \frac{11}{3780}f_{n+\frac{2}{3}} + \frac{332}{2835}f_{n+1} - \frac{269}{3780}f_{n+\frac{4}{3}} + \frac{22}{945}f_{n+\frac{5}{3}} - \frac{37}{11340}f_{n+2}\right)$	8	$\frac{8}{6200145}$
$y_{n+1} = y_n + h\left(\frac{137}{1344}f_n + \frac{27}{56}f_{n+\frac{1}{3}} + \frac{387}{2240}f_{n+\frac{2}{3}} + \frac{34}{105}f_{n+1} - \frac{243}{2240}f_{n+\frac{4}{3}} + \frac{9}{280}f_{n+\frac{5}{3}} - \frac{29}{6720}f_{n+2}\right)$	8	$\frac{1}{653184}$
$y_{n+\frac{4}{3}} = y_n + h\left(\frac{286}{2835}f_n + \frac{464}{945}f_{n+\frac{1}{3}} + \frac{128}{945}f_{n+\frac{2}{3}} + \frac{1504}{2835}f_{n+1} + \frac{58}{945}f_{n+\frac{4}{3}} + \frac{16}{945}f_{n+\frac{5}{3}} - \frac{8}{2835}f_{n+2}\right)$	8	$\frac{8}{6200145}$
$y_{n+\frac{5}{3}} = y_n + h\left(\frac{3715}{36288}f_n + \frac{725}{1512}f_{n+\frac{1}{3}} + \frac{2125}{12096}f_{n+\frac{2}{3}} + \frac{250}{567}f_{n+1} + \frac{3875}{12096}f_{n+\frac{4}{3}} + \frac{235}{1512}f_{n+\frac{5}{3}} - \frac{275}{36288}f_{n+2}\right)$	8	$\frac{275}{158723712}$
$y_{n+2} = y_n + h\left(\frac{41}{420}f_n + \frac{18}{35}f_{n+\frac{1}{3}} + \frac{9}{140}f_{n+\frac{2}{3}} + \frac{68}{105}f_{n+1} + \frac{9}{140}f_{n+\frac{4}{3}} + \frac{18}{35}f_{n+\frac{5}{3}} + \frac{41}{420}f_{n+2}\right)$	9	$-\frac{1}{3061800}$

2.2 Reformulation of $k = 2$ Block Hybrid Linear Multistep Method into Runge – Kutta Type Method

2.2.1: Butcher's Runge - Kutta methods for the first order ordinary differential equations

Muhammad, (2020) defined an s-stage implicit Runge - Kutta methods for fist order ODEs in the form:

$$\left.\begin{array}{l} y_{n+1} = y_n + h\sum_{i=0}^{s} a_{ij}k_i \\ \text{Where } i = 1, 2, 3, \ldots, s \\ k_i = f\left(x_i + \alpha_i h,\ y_n + h\sum_{i=0}^{s} a_{ij}k_i\right) \end{array}\right\} \quad (10)$$

The parameters α_j, k_j and a_{ij} defined the method. In Butcher array form, it can be written as

α	β
	W^T

The Butcher array conditions for first order differential equation is stated as:

(i). $\sum_{j=1}^{s} a_{ij} = c_i$ (ii). $\sum_{j=1}^{s} b_j = 1$. (11)

Source:(Muhammad, 2020)

Hence 9 is modified to satisfy the conditions which gave rise to the Butcher table.

(17)

2.2.2: Butcher's Runge Kutta methods for the second order ordinary differential equations

Consider general second order differential equations (ODEs) with initial value problems (IVP) of the form

$y''(x) = f\big(x, y, y'(x)\big), y(0) = \varphi, y'(0) = \psi$

(18)

Muhammad, (2020) defined an s-stage implicit Runge - Kutta type methods for second order ODEs in the form:

$$\left.\begin{aligned} y_{n+1} &= y_n + \alpha_j h y'_n + h^2 \textstyle\sum_{i=1}^{s} a_{ij} k_i \\ y'_{n+1} &= y'_n + h \textstyle\sum_{i=1}^{s} \overline{a_{ij}} k_i\,, \quad where\ i = 1,2,3, \dots s \\ k_i &= f\left(x_i + \alpha_j h, y_n + \alpha_j h y'_n + h^2 \sum_{i=1}^{s} a_{ij} k_i, y'_n + h \sum_{i=1}^{s} \overline{a_{ij}} k_i \right) \end{aligned}\right\} \quad (13)$$

The parameters α_j, k_j, a_{ij} and $\bar{a}_{ij}$ defined the method. In Butcher array form, it can be written as:

α	$\bar{A}$	A
	$\bar{b}^T$	b

C	A						
0	0	0	0	0	0	0	0
$\frac{1}{6}$	$\frac{19087}{362880}$	$\frac{2713}{15120}$	$\frac{-15487}{120960}$	$\frac{293}{2835}$	$\frac{-6737}{120960}$	$\frac{263}{15120}$	$\frac{-863}{362880}$
$\frac{1}{3}$	$\frac{1139}{22680}$	$\frac{47}{189}$	$\frac{11}{7560}$	$\frac{166}{2835}$	$\frac{-269}{7560}$	$\frac{11}{945}$	$\frac{-37}{22680}$
$\frac{1}{2}$	$\frac{137}{2688}$	$\frac{27}{112}$	$\frac{387}{4480}$	$\frac{17}{105}$	$\frac{-243}{4480}$	$\frac{9}{560}$	$\frac{-29}{13440}$
$\frac{2}{3}$	$\frac{143}{2835}$	$\frac{232}{945}$	$\frac{64}{945}$	$\frac{752}{2835}$	$\frac{29}{945}$	$\frac{8}{945}$	$\frac{-4}{2835}$
$\frac{5}{6}$	$\frac{3715}{72576}$	$\frac{725}{3024}$	$\frac{2125}{24192}$	$\frac{125}{567}$	$\frac{3875}{24192}$	$\frac{235}{3024}$	$\frac{-275}{72576}$
1	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{32}{105}$	$\frac{9}{280}$	$\frac{9}{35}$	$\frac{41}{840}$
b=1	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{34}{105}$	$\frac{9}{280}$	$\frac{9}{35}$	$\frac{41}{840}$

where

$A = a_{ij} = \beta^2, \overline{A} = \overline{a_{ij}} = \beta, \beta = \beta\ell, \overline{b} = w, b = w^T\beta$

Extending (16) to Butcher array for second order ODEs as:

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
$\frac{1}{6}$	$\frac{19087}{362880}$	$\frac{2713}{15120}$	$\frac{-15487}{120960}$	$\frac{293}{2835}$	$\frac{-6737}{120960}$	$\frac{263}{15120}$	$\frac{-86}{36288}$	$\frac{647}{103680}$	$\frac{275}{18144}$	$-\frac{12013}{725760}$	$\frac{9}{560}$	$-\frac{7289}{725760}$	$\frac{323}{90720}$	$-\frac{79}{1451}$
$\frac{1}{3}$	$\frac{1139}{22680}$	$\frac{47}{189}$	$\frac{11}{7560}$	$\frac{166}{2835}$	$\frac{-269}{7560}$	$\frac{11}{945}$	$\frac{-37}{2268}$	$\frac{337}{22680}$	$\frac{299}{5670}$	$-\frac{16}{567}$	$\frac{79}{2835}$	$-\frac{53}{3240}$	$\frac{31}{5670}$	$-\frac{1}{126}$
$\frac{1}{2}$	$\frac{137}{2688}$	$\frac{27}{112}$	$\frac{387}{4480}$	$\frac{17}{105}$	$\frac{-243}{4480}$	$\frac{9}{560}$	$\frac{-29}{1344}$	$\frac{125}{5376}$	$\frac{3}{32}$	$-\frac{171}{8960}$	$\frac{5}{112}$	$-\frac{219}{8960}$	$\frac{9}{1120}$	$-\frac{31}{2688}$
$\frac{2}{3}$	$\frac{143}{2835}$	$\frac{232}{945}$	$\frac{64}{945}$	$\frac{752}{2835}$	$\frac{29}{945}$	$\frac{8}{945}$	$\frac{-4}{283}$	$\frac{2}{63}$	$\frac{76}{567}$	$-\frac{16}{2835}$	$\frac{232}{2835}$	$-\frac{16}{567}$	4 [illegible]	[illegible]
$\frac{5}{6}$	$\frac{3715}{72576}$	$\frac{725}{3024}$	$\frac{2125}{24192}$	$\frac{125}{567}$	$\frac{3875}{24192}$	$\frac{235}{3024}$	$\frac{-27}{7257}$	$\frac{5825}{145152}$	$\frac{3175}{18144}$	$\frac{125}{20736}$	$\frac{125}{1008}$	$-\frac{1625}{145152}$	$\frac{2[illegible]}{18144}$	$-\frac{[illegible]}{1451}$
1	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{32}{105}$	$\frac{9}{280}$	$\frac{9}{35}$	$\frac{41}{840}$	$\frac{41}{840}$	$\frac{3}{14}$	$\frac{3}{140}$	$\frac{17}{105}$	$\frac{3}{280}$	$\frac{3}{70}$	0
	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{32}{105}$	$\frac{9}{280}$	$\frac{9}{35}$		$\frac{41}{840}$	$\frac{3}{14}$	$\frac{3}{140}$	$\frac{17}{105}$	$\frac{3}{280}$	$\frac{3}{70}$	
	$\frac{41}{840}$							0						

(20)

The (18) Butcher array satisfied the conditions for second order differential equations which is:

$(i).\ \sum_{j=1}^{s} b_j = \frac{1}{2}.$ Source: (Muhammad, 2020)

From (21) seven stage implicit Ruge-Kutta type method was obtained as (21) – (29):

$$y_{n+1} = y_n + hy' + h^2\left(\frac{41}{840}k_1 + \frac{3}{14}k_2 + \frac{3}{140}k_3 + \frac{17}{105}k_4 + \frac{3}{280}k_5 + \frac{3}{70}k_6\right)$$
(21)

$$y'_{n+1} = y' + h\left(\frac{41}{840}k_1 + \frac{9}{35}k_2 + \frac{9}{280}k_3 + \frac{34}{105}k_4 + \frac{9}{280}k_5 + \frac{9}{35}k_6 + \frac{41}{840}k_7\right)$$
(22)

$$k_1 = f(x_n, y_n, y')$$
(23)

$$k_2 = f\left(x_n + \frac{1}{6}h, y_n + \frac{1}{6}hy'_n + h^2\left(\frac{647}{103680}k_1 + \frac{275}{18144}k_2 - \frac{12013}{725760}k_3 + \frac{9}{560}k_4 - \frac{7289}{725760}k_5 + \frac{323}{90720}k_6 - \frac{79}{145152}k_7\right), y' + h\left(\frac{19087}{362880}k_1 + \frac{2713}{15120}k_2 - \frac{15487}{120960}k_3 + \frac{293}{2835}k_4 - \frac{6737}{120960}k_5 + \frac{263}{15120}k_6 - \frac{863}{362880}k_7\right)\right) \quad (24)$$

$$k_3 = f\left(x_n + \frac{1}{3}h, y_n + \frac{1}{3}hy'_n + h^2\left(\frac{337}{22680}k_1 + \frac{299}{5670}k_2 - \frac{16}{567}k_3 + \frac{79}{2835}k_4 - \frac{53}{3240}k_5 + \frac{31}{5670}k_6 - \frac{1}{1260}k_7\right), y' + h\left(\frac{1139}{22680}k_1 + \frac{47}{189}k_2 + \frac{11}{7560}k_3 + \frac{166}{2835}k_4 - \frac{269}{7560}k_5 + \frac{11}{945}k_6 - \frac{37}{22680}k_7\right)\right)$$
(25)

$$k_4 = f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hy'_n + h^2\left(\frac{125}{5376}k_1 + \frac{3}{32}k_2 - \frac{171}{8960}k_3 + \frac{5}{112}k_4 - \frac{219}{8960}k_5 + \frac{9}{1120}k_6 - \frac{31}{26880}k_7\right), y' + h\left(\frac{137}{2688}k_1 + \frac{27}{112}k_2 + \frac{387}{4480}k_3 + \frac{17}{105}k_4 - \frac{243}{4480}k_5 + \frac{9}{560}k_6 - \frac{29}{13440}k_7\right)\right)$$
(26)

$$k_5 = f\left(x_n + \frac{2}{3}h, y_n + \frac{2}{3}hy'_n + h^2\left(\frac{2}{63}k_1 + \frac{76}{567}k_2 - \frac{16}{2835}k_3 + \frac{232}{2835}k_4 - \frac{16}{567}k_5 + \frac{8}{405}k_6 - \frac{4}{2835}k_7\right), y' + h\left(\frac{143}{2835}k_1 + \frac{232}{945}k_2 + \frac{64}{945}k_3 + \frac{752}{2835}k_4 + \frac{29}{945}k_5 + \frac{8}{945}k_6 - \frac{4}{2835}k_7\right)\right)$$
(27)

$$k_6 = f\left(x_n + \frac{5}{6}h, y_n + \frac{5}{6}hy'_n + h^2\left(\frac{5825}{145152}k_1 + \frac{3175}{18144}k_2 + \frac{125}{20736}k_3 + \frac{125}{1008}k_4 - \frac{1625}{145152}k_5 +\right.\right.$$

$\frac{275}{18144}k_6 - \frac{275}{145152}k_7\Big), y' + h\Big(\frac{3715}{72576}k_1 + \frac{725}{3024}k_2 + \frac{2125}{24192}k_3 + \frac{125}{567}k_4 + \frac{3875}{24192}k_5 + \frac{235}{3024}k_6 - \frac{275}{72576}k_7\Big)\Big)$ (28)

$k_7 = f\Big(x_n + \frac{5}{6}h, y_n + \frac{5}{6}hy'_n + h^2\Big(\frac{41}{840}k_1 + \frac{3}{14}k_2 + \frac{3}{140}k_3 + \frac{17}{105}k_4 + \frac{3}{280}k_5 + \frac{3}{70}k_6\Big), y' + h\Big(\frac{41}{840}k_1 + \frac{9}{35}k_2 + \frac{9}{280}k_3 + \frac{32}{105}k_4 + \frac{9}{280}k_5 + \frac{9}{35}k_6 + \frac{41}{840}k_7\Big)\Big)$
(29)

3.0 Findings and Discussion

To verify the credibility of our proposed scheme, some problems were selected for testing and the results were compared with other existing literature.

Test Problem 1: $\mu''(t) = -1001\mu' - 1000\mu$, $h = 0.05$; $\mu(0) = 1$, $\mu'(0) = -1$.
Exact solution is $\mu(t) = e^{-t}$ (Akinnukawe and Okunuga, 2024)

Test Problem 2:

$$\mu'' = \mu', \mu(0) = 1, \mu'(0) = \frac{1}{2}, h = \frac{1}{100}, 0 \le x \le 1$$

Exact solution: $y(x) = 1 - e^t$ (Akinnukawe and Okunuga, 2024)

Table 3.1: Comparison of the result between exact solution, Runge-Kutta type method and Akinnukawe and Okunuga, (2024) for test problem 1

x	Exact Solution	Proposed Method	Error in Proposed Method	Error in Akinnukawe and Okunuga, (2024)
0.1	0.95122942450071400910	0.95122942450071400909	1. 00E -20	1.85296E-13
0.2	0.90483741803595957318	0.90483741803595957316	2. 00E -20	3.35731E-13
0.3	0.86070797642505780725	0.86070797642505780723	2. 00E -20	4.54636E-13
0.4	0.81873075307798185870	0.81873075307798185867	3. 00E -20	5.48339E-13
0.5	0.77880078307140486828	0.77880078307140486825	3. 00E -20	6.19726E-13
0.6	0.74081822068171786611	0.74081822068171786607	4. 00E -20	6.72684E-13
0.7	0.70468808971871343440	0.70468808971871343435	5. 00E-20	7.09821E-13
0.8	0.67032004603563930079	0.67032004603563930074	5. 00E -20	7.34135E-13

Table 3.1 showed the absolute errors $|Y(x_n) - y_n|$ of the Runge-Kutta Type method and Akinnukawe and Okunuga, (2024) on test problem 1 for $h = 0.05$, Clearly the table shows that the Runge-Kutta type method closer to the exact than Akinnukawe and Okunuga, (2024) . The Runge-Kutta Type method approaches the exact solution faster than Akinnukawe and Okunuga, (2024) within the interval of $0 \le x \le 1$ for $h = 0.05$ on test problem 1

Table 3.2: Comparison of the result between exact solution, Runge Kutta Type method and Akinnukawe and Okunuga, (2024) for test problem 2

x	Exact Solution	Proposed Method	Error in the proposed method	Error in Akinnukawe and Okunuga, (2024)
0.01	-0.0100501670841681	-0.010050167084168058	4.2 E-17	3.10862E-15
0.02	-0.0202013400267558	-0.020201340026755811	1.1E-17	1.39888E-14
0.03	-0.0304545339535169	-0.030454533953516857	4.3E-17	3.24185E-14
0.04	-0.0408107741923882	-0.040810774192388229	2.9E-17	6.12843E-14
0.05	-0.0512710963760240	-0.051271096376024043	4.3E-17	1.01918E-13
0.06	-0.0618365465453596	-0.061836546545359626	2.6E-17	1.56541E-13

0.07	-0.0725081812542165	-0.072508181254216484	1.6E-17	2.28706E-13
0.08	-0.0832870676749586	-0.083287067674958561	3.9E-17	3.19744E-13
0.09	-0.0941742837052104	-0.094174283705210366	3.4E-17	4.33875E-13
0.1	-0.1051709180756476	-0.10517091807564763	3.0E-17	5.75096E-13

Table 3.2 showed the absolute errors $|Y(x_n) - y_n|$ of the Runge-Kutta type method and Akinnukawe and Okunuga, (2024) on test problem 1 for $h = 0.01$. Clearly the table shows that the Runge-Kutta type method closer to the exact than Akinnukawe and Okunuga, (2024). The Runge-Kutta type method approaches the exact solution faster than Akinnukawe and Okunuga, (2024) within the interval of $0 \leq x \leq 0.1\ for\ h = 0.01$ on test problem 2

4.0 Conclusion

Reformulation of the two-step implicit linear multi-step block hybrid method into a Runge–Kutta-Type scheme for the solution of second-order initial value problems (IVPs) in ordinary differential equations represents an important advancement in numerical methods for the solution of ordinary differential equations. The reformulated method propagates the efficiency and stability properties of the hybrid method it was reformulated from while incorporating the adaptability and self-starting type of Runge–Kutta schemes. The Runge–Kutta-Type method derived demonstrates uniform accuracy and computational efficiency because of its better performance in numerical experiments when compared to the existing methods in the literature. The method’s capability to handle second-order ODEs without reduction provides a versatile tool for wide applications in science and engineering.

5.0 Recommendation

In view of the results obtained, it is recommended that advance research be conducted to extend the applicability of this method. Future studies could investigate its performance on its adaptation to variable step-size techniques for enhanced computational efficiency. Also, investigating its application to higher-order ordinary differential equations and large-scale systems would further validate its flexibility and reliability. The method's strength for parallel implementation should be examined too to support modern computational resources effectively. In summary, the proposed Runge–Kutta-Type method gives an unquantifiable contribution to the field of numerical analysis and holds great promise for solving advanced initial value problems.

REFERENCES

Akinnukawe, B. I. & Okunuga, S. A. (2024). One-step block scheme with optimal hybrid points for numerical integration of second-order ordinary differential equations. *Journal of Nigerian Society of Physical Sciences*, *6*(2).

Datsko, B. & Kutniv, M. (2024). Explicit numerical methods for solving singular initial value problems for systems of second-order nonlinear ODEs. *Numerical Algorithms*, 1-14.

Denis, B. (2020). An overview of numerical and analytical methods for solving ordinary differential equations. *arXiv preprint arXiv:2012.07558*.

Li, J., Cui, N., Zhu, Y., & Yu, K. (2024). Novel Implicit Integration Algorithms With Identical Second-Order Accuracy and Flexible Dissipation Control for First-Order Transient Problems. *International Journal for Numerical Methods in Engineering*, e7639.

Muhammad, R. (2020). The Order and Error Constant of a Runge-Kutta Type Method for the Numerical Solution of Initial Value Problem.

Olaleye, A. O. & Fiyinfoluwa, A. (2023). Solution of Second-Order Ordinary Differential Equations Through One-Step Hybrid Block Method.

Omole, E. O., Adeyefa, E. O., Emadifar, H., Abualigah, L., Onu, P., Ajewole, K. P., Adeniji, T. O., Okoro, J. O., & Amoyedo, F. E. (2024). Development and Implementation of a Top-order Multi-step Block Method for Boundary Value Problems: The advancement of numerical accuracy. 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG),

Sharma, P. L. (2021). Euler's Method For Solving Initial Value Problems In Ordinary Differential Equations.

Taıwo, O. E., Moshood, L. O., & Ogunnıran, M. O. (2023). Effectiveness of a One-fifth Hybrid Block Approach for Second Order Ordinary Differential Equations. *Istanbul Journal of Mathematics*, *1*(2), 86-95.